AS1
Analog Subsystem Manual

MICROWARE SYSTEMS CORPORATION
P.O. BOX 954 • DES MOINES, IOWA 50304 • (515) 279-9856

AS-1 ANALOG SUBSYSTEM MANUAL

MICROWARE SYSTEMS CORPORATION

Comments regarding Microware products and documentation are most appreciated, and help us improve present and future products. If you have any comments regarding this product, please be sure to write or call.

Copyright (C) 1977 Microware Systems Corporation
All rights reserved.

This manual was prepared and edited entirely on a M6800 system using the Microware RT/68 real time operating system and editor.

Part number AS1M

GENERAL INFORMATION

The Microware AS-1 is an input/output interface circuit desisned for M6800 computer systems usins the SS-50 bus. It sives the M6800 capabilities for performins analos-to-disital and disital-to-analos conversions.

Applications include process and machine control, audio and waveform synthesis, sensor input/output, sraphic displays, automated testins systems, etc.

The heart of the AS-1 is the Motorola MC1408L8 monolithic 8-bit D/A converter IC. This relatively low cost/high performance circuit is used in the AS-1 for both A/D and D/A functions. The A/D techniques used in the AS-1 use the microcomputer MPU to perform the conversion control which increases the cost/effectiveness of the AS-1.

This manual also contains applications information and software for a variety of potential applications to provide both examples and data on uses and operation of the AS-1.

SPECIFICATIONS

DIGITAL TO ANALOG OPERATION:

Output Voltase Ranse	0.0 to 6.375 V
Bit Increment Value	0.025 V
Accuracy	0.25% (1/2 LSB)
Settlins Time	4 usec max.
	2 usec typ.

ANALOG TO DIGITAL OPERATION:

Input Voltase (Max for conversion)	6.375 V
Input Voltase (Absolute max.)	8.5 V
Accuracy	0.5% (1 LSB)
Conversion time	243 MPU cycles
Sample-and-hold droop	2 mv/ms.
Sample-and-hold aquisition time	80 us @ 100 ohm drivins Z

GENERAL:

Temperature Ranse	0 to 70 C
Power Supply	+7-9 V @ 150 ma.
	-11.5 to 16 V @ 25 ma.
	+11.5 to 16 V @ 25 ma.

AS-1 OPERATION AND PROGRAMMING

Initialization

The AS-1's PIA must be properly initialized before any conversion is attempted. This may be performed once at power-up or system reset. The example below assumes a port address of $8008, as will all other programming examples shown. This routine will initialize the PIA correctly:

```
   INXAS1 LDA A #$FF       CONSTANT FOR ALL OUTPUTS
          STA A $8008      SET "A" SIDE DATA DIRECTION REGISTER
          LDA A #$0F       CONST. FOR BITS 0-3 OUTPUTS, 4-7 INPUTS
          STA A $800A      SET "B" SIDE DATA DIRECTION REGISTER
          LDA A #$34       CONTROL WORD
          STA A $8009      SET CONTROL REGISTER A
          STA A $800B      SET CONTROL REGISTER B
          RTS              RETURN (USE IF REQUIRED)
```

D/A Conversion

The D/A conversion process is extremely straightforward: the PIA "A" data register is set to the desired value. The voltage output of the DAC is 25 millivolts (0.025 V) times the digital byte stored. The greatest output value possible is 6.375 V which is .025 times 255. For example, the instructions to output 4 volts are:

```
    LDA A #160  LOAD ACC WITH 4/0.025
    STA A $8008
```

In this example the output is a constant, but in most cases the data to be converted will be obtained from a calculation or table. The example below will output a series of values from a table every 1 ms. The index register is used to access the data.

```
CNVTBL  LDX   #TBLBEG   LOAD XR WITH START ADDR OF TABLE
CNV2    LDA A 0,X       LOAD ACC WITH DATA
        STA A $8008     STORE DATA AT DAC ADDRESS (CONVERT)
        LDA B #DELAY    LOAD ACC WITH DELAY CONSTANT (DEPENDS ON CLOCK)
CNVDEL  DEC B           DECREMENT DELAY COUNT
        BNE   CNVDEL    BRA IF DELAY NOT FINISHED
        INX             ADVANCE POINTER TO DATA
        CPX   #TBLEND+1 CHECK IF END OF TABLE
        BNE   CNV2      LOOP IF NOT
        RTS             FINISHED
```

The value of the constant "DELAY" depends on the system clock. The loop execution time is 23+6*D MPU cycles where D is the value of DELAY. This type of routine is particularly useful in waveform synthesis where a series of voltages must be generated at regular intervals.

HARDWARE INTERFACE CONSIDERATIONS

A/D Input Interfacing

The analog inputs are directly connected to the CMOS analog multiplexer so several precautions must be taken:

1. no voltage greater than +8 volts may be applied.
2. no voltage less than -1 volt may be applied
3. if present, static and transient voltages must be controlled.

If necessary, clamping diodes between the inputs and ground/+9V supply may be required.

Input connections are best made with high-quality shielded cable which should be grounded at both ends. The driving impedance should be as low as practicable if the sample and hold is to be used and the aquisition time is critical. The input impedance of the AS-1 is approximately 1 megohm.

D/A Output Interfacing

The AS-1 has a low-impedance operational amplifier output that can drive most user devices directly (about 75 ohms). If noise pickup can be a problem (driving a hi-Z input) use shielded cable grounded at both ends.

GRAPHIC DISPLAYS

If you have two AS-1s and an oscilloscope you have an instant graphics display system. Connect the output of each AS-1 to the scope's X (horiz.) and Y (vert.) inputs and you can position the beam anywhere on the 256 by 256 "grid" on the scope's screen. The longer the spot is held, the brighter it appears. There are may other graphics schemes the AS-1 may be adapted to, most notably stroke (vector) written displays if a constant-writing speed circuit is added.

APPLICATIONS INFORMATION SOURCES

We highly recommend:

SERENDIPITOUS CIRCLES by D Anderson and W Galway, BYTE Magazine, August '77 - FANTASTIC graphics/animation!

INTERFACING TO AN ANALOG WORLD by J Carr BYTE Magazine, May '77.

ANALOG-TO-DIGITAL CONVERSION WITH THE M6800 MICROPROCESSOR- Application Note AN-757, Motorola, Inc. A good source for DVM-type information. Ap notes can be usually had at no cost from your helpful local Motorola sales office.

Finally, the Motorola data sheets on the MC1408 contain a wealth of information on polarity switching systems, applications, etc. It may be found in the Motorola Linear data book.

Analog-to Digital Conversion

The AS-1 uses a hardware/software conversion method which provides considerable economy and performance advantages. The hardware lends itself to several methods for software A/D conversions, but the fastest and most efficient in terms of program size vs. speed is a technique called SUCCESSIVE APPROXIMATION.

The AS-1 is configured so that an unknown analog input selected by the analog multiplexer can be compared against the output of the DAC system. The comparator provides a digital level that indicates the relative magnitude and may be sampled by the MPU via bit PB7 of the PIA. The conversion software determines the incoming signal's voltage by outputing voltages via the DAC and checking the comparator's output to establish a new trial output. After several trials the incoming voltage will be within one bit of the DAC output and is known.

The successive approximation system uses a series of trial values, each of which are half the previous value, to output to the DAC. If the comparator indicates that the current trial value when added to the DAC output is higher than the unknown value, the trial number is subtracted from the DAC output number. After 8 trials (for an 8-bit converter such as the AS-1) the DAC will have a value within 1 bit of the unknown voltage.

The complete algorithm for successive approximation is:

1. Set the DAC to 0
2. Set TRIAL to 128
3. Add TRIAL to DAC
4. If the comparator output is 0 (DAC<X) go to step 6
5. Subtract TRIAL from DAC (restore old value)
6. Divide TRIAL by 2
7. If TRIAL is greater than or equal to 1 go to step 3
8. End - DAC is result

Assume an input (unknown) voltage of 5.28 volts. The table below lists all the intermediate results of the use of the above algorithm.

DAC#	DAC V.	Trial#	New DAC#	New DAC V.	Comp.	Operation
0	0	128	128	3.2	0	continue
128	3.2	64	192	4.8	0	continue
192	4.8	32	224	5.6	1	sub. 32
192	4.8	16	208	5.2	0	continue
208	5.2	8	216	5.4	1	sub. 8
208	5.2	4	212	5.3	1	sub. 4
208	5.2	2	210	5.25	0	continue
210	5.25	1	211	5.275	0	end

The final value of the DAC is 210 or 5.275V divided by .025 volts which is the bit increment value of the DAC. This is the closest number to the actual value of the input voltage that can be determined.

The implementation of the algorithm for the M6800 is simple and short due to the fact that the TRIAL value can be calculated by simple shifting (a binary right shift is a division by two) and all data can be kept in the accumulators. The basic successive approximation program, written as a subroutine, is listed below:

```
ADCNV   CLR A           ACC A WILL CONTAIN INTERMEDIATE RESULT
        CLR B           ACC B WILL CONTAIN TRIAL NUMBER
        SEC             SET CARRY TO SHIFT IN 10000000 BINARY
ADLOOP  ROR B           DIVIDE TRIAL BY 2
        BNE ADCNV2      IF TRIAL IS ZERO CONVERSION IS DONE
        RTS             RETURN WHEN FINISHED....
ADCNV2  ABA             ADD TRIAL TO TEMP RESULT
        STA A $8008     GIVE IT TO THE DAC
        TST   $800A     CHECK COMP. OUTPUT (SIGN BIT PB7)
        BPL   ADLOOP    BRA IF COMPARATOR OUT IS 0
        SBA             SUBTRACT TRIAL FROM TEMP
        BRA   ADLOOP    CONTINUE...
```

An analysis of this program shows that the conversion requires 11 MPU cycles for instructions executed only once (the first three and the RTS), plus 23 or 29 MPU cycles for each of eisht passes throush the loop, dependins on the comparator state. The worst case conversion time is therefore 243 MPU cycles. The program requires 19 bytes of RAM or ROM and is relocatable. If the application requires constant conversion rates, the BPL instruction can branch to a NOP and BRA to even the time to 29 cycles every pass.

Faster conversions can be made at the expense of less precision by limitins the number of conversion loops. This can be done by addins an AND B instruction after the ROR B to truncate TRIAL to the desired number of bits. This will add 2 MPU cycles to each pass but reduce the total number of passes. A six bit conversion, for example, requires 197 MPU cycles.

The routine can be called from a subroutine desisned to input a continuous series of data to be saved for later processins. The routine below uses the index resister to point to an area of memory where the data is to be stored. The number of values to be saved depends on the value of COUNT. BUFBEG is the besinnins address of the buffer.

```
INVLTS  LDX   #BUFBEG   INITIALIZE DATA POINTER
INV2    BSR   ADCNV     CALL CONVERSION ROUTINE
        STA A 0,X       STORE RESULT OF CONVERSION
        INX             ADVANCE POINTER
        DEC   COUNT     DECREMENT COUNTER
        BNE   INV2      LOOP IF NOT FINISHED
        RTS             DONE
```

Usins the program above, sample rates of around 3,500/sec. can be obtained. This speed can be further improved by buildins the conversion program into the loop instead of callins it as a subroutine.

SAMPLE-AND-HOLD FUNCTION

The sample-and-hold is included because some signals that are to be converted may not be stable enough for accurate conversion. The successive approximation will not give accurate data if the input changes significantly during conversion (more than about 10 mv.).

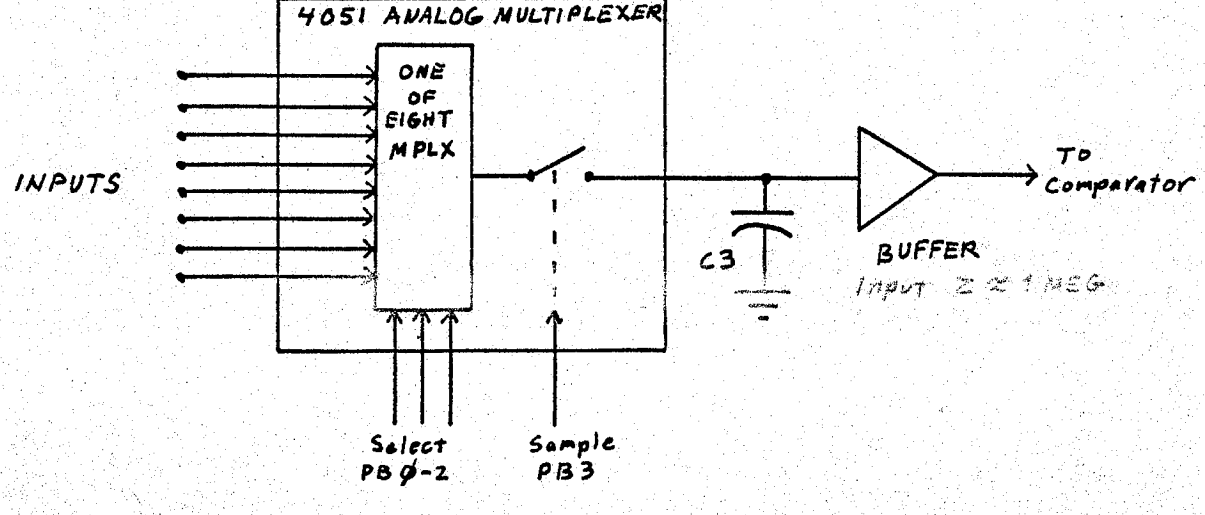
The 4051 multiplexer has an inhibit input that isolates the output from all inputs when asserted. Because the level translators invert the logic level, PIA output PB3 inhibits the multiplexer when it has a 0 output. The multiplexer output drives a low-leakage polycarbonate capacitor (C3) which will charge to the level of the selected input signal. The charge is held when PB3 is 0 (inhibit) and can only discharge slowly through the high impedance of the buffer op-amp. It does, however, retain its charge long enough to allow an accurate conversion of the "captured" input voltage.

The time that must be allowed for the charging process of C (sample) ("aquisition time") depends on the impedance of the input voltage source. For a 10 ohm impedance, the time is about 20 microseconds, and increases to about 80 microseconds for a 1000 ohm impedance. The output impedance of typical 741 operational amplifiers is about 10 to 100 ohms.

The decay rate of the sample and hold is about 2 millivolts per millisecond, which is quite adequate for successive approximation conversion.

The short routine shown below combines sample-and-hold control with input multiplexer selection. The subroutine is called with the desired input number in accumulator A:

```
SMPHLD ORA A #$08       TURN ON BIT 3
       STA A $800A      STORE IT AT ADC PIA ADDRESS
       LDA B #10        DELAY COUNT FOR ABOUT 80 USEC.
SHDEL  DEC B            DECREMENT DELAY COUNT
       BNE   SHDEL      BRA IF NOT OVER
       AND A #$07       TURN OFF ALL BITS BUT INPUT #
       STA A $800A      STORE AT ADC PIA
       JMP   ADCNV      CONVERT ENDS WITH RTS....
```



PROCESSING THE CONVERTED DATA

The value returned by the successive approximation subroutine is a representation of the input voltage in 25 millivolt (1/40 V) units represented as a binary number. In many cases this format is sufficient for further processing, but in some instances another format may be desired. Some techniques for conversion of raw data are illustrated below.

Conversion to BCD - Digital Voltmeter

The conversion of raw data to BCD form is essentially an implementation of a 2-1/2 digit digital voltmeter system. The byte returned in accumulator A by the A/D conversion routine is transformed to three bytes of BCD information - units, tenths and hundreths. The raw data is 1/40 volt units, so the first step is to divide the data by 40 to obtain the units digit. The division is accomplished by repeated subtraction, and the remainder represents the fraction. Another division by four will yield the tenths digit. The remainer of this division is a number from zero to three 1/40s which is rounded to either zero or five hundreths which is consistent with the conversion's accuracy.

```
00004                    * CONVERT DATA IN ACC A TO BCD FORMAT
00005 0020                       ORG    $20
00006 0020 0001     UNITS  RMB    1
00007 0021 0001     TENTHS RMB    1
00008 0022 0001     HNDTHS RMB    1

00010 0200                       ORG    $200
00011 0200 5F       CNVTD  CLR B           DIVIDE BY 40
00012 0201 81 28    CNVONE CMP A  #40      SEE IF DIVIDEND<40
00013 0203 25 05            BCS    CNVTNT  BRA IF IT IS
00014 0205 80 28            SUB A  #40     SUBTRACT DIVISOR
00015 0207 5C               INC B          INCR QUOTIENT
00016 0208 20 F7            BRA    CNVONE  REPEAT
00017 020A D7 20    CNVTNT STA B  UNITS    SAVE ONES
00018 020C 16               TAB            SAVE REMAINDER IN ACC B
00019 020D 44               LSR A          DIVIDE REM. BY 4
00020 020E 44               LSR A
00021 020F 97 21            STA A  TENTHS  SAVE TENTHS
00022 0211 C4 03            AND B  #%00000011  GET 1/40S
00023                    * ROUND TO 0 OR 5
00024 0213 4F               CLR A
00025 0214 C1 01            CMP B  #1
00026 0216 23 02            BLS    PUTHND  MAKE IT 0 IF REM<2
00027 0218 86 05            LDA A  #5      OTHERWISE 5
00028 021A 97 22    PUTHND STA A  HNDTHS
00029 021C 39               RTS            DONE
00030                       END
```

ASSEMBLY INSTRUCTIONS

I M P O R T A N T ! ! ! !

THIS CIRCUIT USES MOS DEVICES WHICH ARE SUSCEPTABLE TO STATIC DAMAGE FROM IMPROPER HANDLING. READ INSTRUCTIONS CAREFULLY !

1. Refer to the parts list and circuit layout diagram for proper component locations. Make sure all ICs are oriented so pin 1 is in the right place. Note that the 10 uF electrolytic capacitor (C2) must be installed with the correct polarity. The transistors must also be installed properly. Some AS-1s are supplied with the transistors in round epoxy cases. The flat side of the case is the emitter "E" with these.

2. Inspect the printed circuit board for any flaws or broken foils.

3. Install the three connectors ON THE TOP SIDE of the PC board.

4. Install all resistors and capacitors according to the circuit layout diagram. Carefully clean all solder resin around the analog multiplexer and particularly C3. Use denatured alcohol or any non-corrosive electronic solvent such as trichlorethylene.

5. Install and solder all IC's except IC1 and IC5. Carefully check for "solder bridges".

6. IC1 and IC5 are MOS devices and must be carefully handled. Leave them in protective foam until installation. Ground yourself to a GOOD ground through a protective 1 megohm resistor, and ground the soldering iron as well. A suggested method is to use an alligator clip attached to a metallic watchband and to the ground via the 1 megohm resistor (included). Do not wear nylon clothing.

7. Before the AS-1 is installed in the computer system, visually inspect all solder connections and re-check IC and transistor positioning. A small nylon "index pin" has been provided to prevent incorrect insertion of the card. Install it in the connector hole below the V etched on the top side of the P.C.

8. The AS-1 should now be ready for operation. NEVER INSERT OR REMOVE CIRCUIT BOARD WITH POWER ON.

CHECKOUT PROCEDURE AND DIAGNOSTICS

1. Circuit voltage checks - using a VOM, VTVM, DVM, etc., observe the voltages on the following IC pins for correct voltages:

IC4 - PIN 4	-9.0 V +/- 0.5V
IC4 - PIN 8	+9.0 V +/- 0.5V
IC1 - PIN 20	+5.0 V +/- 0.025V

CHECKOUT AND DIAGNOSTICS - CONT'D

If any of the listed voltages are not within parameters, remove system power and re-check for shorts, etc. Do not proceed until the readings are correct.
(TESTS POWER SUPPLY SYSTEMS)

2. Turn on the computer system and install the AS-1 in I/O port number 2 (third from the left, next to the control interface). Use the systems monitor to examine memory locations $8008, $8009, $800A and $800B. All four should read 00. If they all read $FF, the AS-1 is probably in the wrong I/O slot. If the locations are non-zero but not $FF, try a system reset and repeat this step. If this fails, a defective or static damaged PIA is indicated.
(TESTS PIA AND BUS INTERFACE)

3. Store the hex numbers $FF and $34 respectively in memory locations $8008 and $8009. You may get a no-change error after the second store, disregard it. Open memory location $8008 and make sure it is 00. Connect a voltmeter or scope to the analog output of the AS-1. The voltage should read within a few millivolts of 0. Store the value $FF at address $8008. The voltage at the output should read from 5-8 volts. Adjust R3 for about 6.5 volts. If this step fails twice, check components around and including ICs 2 and 3.
(TESTS DIGITAL-TO-ANALOG SYSTEM)

4. Store the hex numbers $0F and $34 respectively in memory locations $800a and $800B, again ignoring a no-change error if necessary. Set $8008 to $FF. Run a jumper from analog input #0 to a +5 volt (make sure it's +5!!!) source. Set $800A to $08, you may get a no-change error. Read $800A again and check for a contents of $88. Go back to $8008 and store a $00, and check $800A again. It should read $08 - if not there is a problem in the comparator (IC4), the analog multiplexer (IC5), or the dual op-amp buffer side (IC3) or related circuitry.
(TESTS A/D SYSTEM COMPONENTS)

GENERAL NOTE - IF YOU HAVE DIFFICULTY OBTAINING THE CORRECT PIA (MEMORY) VALUES IN STEPS 2, 3 OR 4 TRY A SYSTEM RESET AND GO BACK TO STEP 2. INITIALIZING THE PIA AND THE 'BURIED' DATA DIRECTION REGISTERS CAN BE TRICKY !

5. Proceed to the "Alignment Procedure" on the next page.

ALIGNMENT AND CALIBRATION

OPTION 1 - ALIGNMENT WITH ACCURATE VOLTMETER

This method requires use of a 3-1/2 digit digital voltmeter or equivalent. Use the monitor's memory examine/change function to set memory locations (PIA addresses) $8008 and $8009 to $FF and $34 respectively. Re-open address $8008 and store a $FF. This should be performed after a system reset. For best temperature stability allow a 15 minute warmup before proceeding. Connect the DVM to the AS-1 analog output and adjust R3 for exactly 6.375 volts. No further alignment is required. You may wish to "seal" R3 with a light drop of rubber cement.

OPTION 2 - ALIGNMENT BY VOLTAGE REFERENCE

This method is a second choice, and uses the most accurate voltage reference available. A laboratory standard is the first choice but any other accurate voltage source may be used. The more stable and accurate the reference is the more accurate the AS-1 will be. A "poor man's lab standard" may be constructed by obtaining a one-cell watch or hearing aid mercury battery and connecting a 100K resistor across it. This will deliver within a few milli-volts of 1.35 volts.

Load the digital voltmeter program listed in this manual and connect the analog input to the voltage source. Use the DVM program while adjusting R3 until the program displays the exact value of the voltage reference. This usually requires several cycles. There is a "range" in R3's rotation where the same reading is obtained. Set R3 to the center of this range.

JUMPER-SELECTABLE OPTIONS

OPTION A - SAMPLE/HOLD CONTROL

If the jumper from point C to point B (above R7) is moved to point A the PIA's control register will determine the S/H state via output CB2 instead of PB7. This may prove useful for separating the input selection-S/H control, or holding the S/H control "on".

OPTION B - INTERRUPTING LEVEL DETECTOR

The AS-1 can be used as a interrupt generating threshhold detector by enabling the IRQ via jumpers (just above the connectors). The comparator output is also connected to control input CA1 of the PIA which may be programmed to detect positive or negative going level changes and causing an interrupt. This will occur when the selected input voltage "passes" the DAC output voltage.

```
00001                           NAM     DIGIVOLT
00002                           OPT     O

00004                   * DIGITAL VOLTMETER PROGRAM FOR MICROWARE CORP.
00005                   *          AS-1 ANALOG SUBSYSTEM
00006                   *
00007                   *   WRITTEN BY: KEN KAPLAN  JULY 1977
00008                   *
00009                   *      (C) COPYRIGHT 1977
00010                   *    MICROWARE SYSTEMS CORPORATION
00011                   *
00012                   * THIS PROGRAM USES THE CONSOLE MONITOR ROM
00013                   * (EITHER MIKBUG (TM) OR RT/68MX (TM) FOR
00014                   * INPUT/OUTPUT SUBROUTINES. THE PROGRAM WILL
00015                   * PRINT A ? PROMPT AND THE USER ENTERS THE
00016                   * NUMBER (O-7) OF THE DESIRED INPUT CHANNEL.
00017                   *
00018                   * THE PROGRAM THEN PERFORMS THE S/A CONVERSION
00019                   * USING THE SAMPLE-AND-HOLD  AND PRINTS THE
00020                   * VOLTAGE IN DECIMAL.


00022                   * DEFINE MEMORY AND MONITOR REFEENCES

00024 0020                      ORG     $20
00025 0020 0001         UNITS   RMB     1       ONES DIGIT
00026 0021 0001         TENTHS  RMB     1       1/10S DIGIT
00027 0022 0001         HNDTHS  RMB     1       1/100S DIGIT

00029       E1AC        INCH    EQU     $E1AC   INPUT CHAR ROUTINE
00030       E07E        PDATA   EQU     $E07E   PRINT DATA STRING

00032       8008        PIADAC  EQU     $8008   AS-1 DAC OUTPUT REG
00033       8009        PIACRA  EQU     $8009   AS-1 CONTROL REG A
00034       800A        PIAADC  EQU     $800A   AS-1 ADC DATA REG
00035       800B        PIACRB  EQU     $800B   AS-1 CONTROL REG B


00037                   * ENTRY POINT ADDRESS $0100
00038 0100                      ORG     $100
00039 0100 86 FF        ENTRY   LDA A   #$FF
00040 0102 B7 8008              STA A   PIADAC  SET DAC REG FOR OUTPUTS
00041 0105 86 0F                LDA A   #$0F
00042 0107 B7 800A              STA A   PIAADC  SET DAC FOR IN/OUT
00043 010A 86 34                LDA A   #$34    CONTROL WORD
00044 010C B7 8009              STA A   PIACRA
00045 010F B7 800B              STA A   PIACRB  INIT. FINISHED

00047                   *MAIN PROGRAM LOOP
00048 0112 CE 017D      MAIN    LDX     #CRLF   LOAD XR W/CR+LF STRING
00049 0115 BD E07E              JSR     PDATA   PRINT IT
00050 0118 86 3F                LDA A   #'?
00051 011A 8D 5E                BSR     OUTCH   PRINT PROMPT
00052 011C 8D 5A                BSR     OUTSP   PRINT SPACE
```

```
00053 011E 8D 5A          BSR    OUTCH
00054 0120 BD E1AC        JSR    INCH       INPUT CHANNEL #
00055 0123 84 07          AND A  #$7        MASK OUT ALL BUT #
00056 0125 8A 08          ORA A  #$08       TURN ON SAMPLE BIT
00057 0127 B7 800A        STA A  PIAADC     STORE AT ADC REG
00058 012A C6 80          LDA B  #$80       LOAD DELAY COUNT
00059 012C 5A       SHDEL DEC B             DECR DELAY COUNT
00060 012D 26 FD          BNE    SHDEL      BRA IF NOT OVER
00061 012F 7F 800A        CLR    PIAADC     SET TO HOLD MODE

00063               * SUCCESSIVE APPROX. CONVERSION
00064 0132 4F             CLR A             CLREAR DAC
00065 0133 5F             CLR B             CLEAR TRIAL
00066 0134 0D             SEC
00067 0135 56       ADLOOP ROR B            DIVIDE TRIAL BY 2
00068 0136 27 0C          BEQ    CNVOVR     BRA IF FINISHED
00069 0138 1B             ABA               ADD TRIAL TO DAC
00070 0139 B7 8008        STA A  PIADAC     PUT IT OUT
00071 013C 7D 800A        TST    PIAADC     TEST COMPARATOR
00072 013F 2A F4          BPL    ADLOOP     BRA IF STILL SMALLER
00073 0141 10             SBA               SUBTRACT TRIAL
00074 0142 20 F1          BRA    ADLOOP

00076               * CONVERT RESULT TO DECIMAL VOLTS
00077               * FIND ONE'S DIGIT
00078 0144 5F       CNVOVR CLR B            FORM ONES IN B
00079 0145 81 28    ONES   CMP A  #40       IS ACCA <40
00080 0147 25 05          BCS    FRAC       BRA IF YES
00081 0149 80 28          SUB A  #40        SUBTRACT 40
00082 014B 5C             INC B
00083 014C 20 F7          BRA    ONES
00084 014E D7 20    FRAC   STA B  UNITS      SAVE ONES DIGIT
00085 0150 16             TAB               COPY REMAINDER
00086 0151 54             LSR B             DVD BY 4
00087 0152 54             LSR B
00088 0153 D7 21          STA B  TENTHS     SAVE TENTHS DIGIT
00089 0155 84 03          AND A  #$03       EXTRACT REMAINDER
00090               *ROUND 1/100S TO 0 OR 5
00091 0157 5F             CLR B
00092 0158 81 01          CMP A  #1         COMPARE TO 1
00093 015A 23 02          BLS    PUTHUN
00094 015C C6 05          LDA B  #5
00095 015E D7 22    PUTHUN STA B  HNDTHS    SAVE HUNDRETHS DIGIT

00097               * AT THIS POINT THE BCD VOLTAGE IS
00098               * CONVERTED AND STORED AND MAY BE
00099               * USED FOR OTHER PROCESSING

00101               * OUTPUT DATA
00102 0160 8D 16          BSR    OUTSP      PRINT SPACE
00103 0162 96 20          LDA A  UNITS      GET ONES DIGIT
00104 0164 8D 0E          BSR    OUTDIG     PRINT IT
00105 0166 86 2E          LDA A  #'.        PRINT DEC. PT.
```

```
00106 0168 8D 10          BSR    OUTCH
00107 016A 96 21          LDA A  TENTHS
00108 016C 8D 06          BSR    OUTDIG
00109 016E 96 22          LDA A  HNDTHS
00110 0170 8D 02          BSR    OUTDIG
00111 0172 20 9E          BRA    MAIN       RETURN TO BEGINNING

00113                * SUBR. TO PRINT DIGIT
00114 0174 8A 30   OUTDIG ORA A  #$30       MAKE IT ASCII
00115 0176 20 02          BRA    OUTCH      PRINT + RTS

00117                * SUBR TO PRINT SPACE
00118 0178 86 20   OUTSP  LDA A  #$20
00119 017A 7E E1D1 OUTCH  JMP    $E1D1      PRINT + RTS
00120 017D 0D      CRLF   FCB    $0D,$0A,0,0,0,0,4
      017E 0A
      017F 00
      0180 00
      0181 00
      0182 00
      0183 04
00121                     END

TOTAL ERRORS 00000
```

```
00001                    NAM    CNVTB



00003             * CONVERT RAW DATA FROM A/D CONV. IN ACC A
00004             * TO MILLIVOLTS IN BINARY IN ACC A, B
00005             * BY MULTIPLYING BY 25.  MULTIPLICATION
00006             * USES FAST SHIFT-AND-ADD SYSTEM
00007             * RESULT IN ACC B (HI) AND A (LO)

00009 0020                ORG    $20
00010 0020 0001   TMPLO   RMB    1          TEMPORARY FOR MULT.
00011 0021 0001   TMPHI   RMB    1

00013 0200                ORG    $200
00014             * ACC A HAS RAW DATA
00015 0200 5F     CNVTMV  CLR B              CLEAR HI BYTE
00016 0201 8D 02          BSR    MUL5       MULT ACC A,B BY 5
00017 0203 20 00          BRA    MUL5       MULT BY 5 (5*5=25)

00019 0205 97 20  MUL5    STA A  TMPLO      SAVE LO BYTE
00020 0207 D7 21          STA B  TMPHI      SAVE HI BYTE
00021 0209 48             ASL A
00022 020A 59             ROL B            MULT BY 2
00023 020B 48             ASL A
00024 020C 59             ROL B            MULT BY 2
00025 020D 9B 20          ADD A  TMPLO
00026 020F D9 21          ADC B  TMPHI     N*5=(N*2)*2+N
00027 0211 39             RTS
00028                     END

TOTAL ERRORS 00000
```

CIRCUIT DESCRIPTION

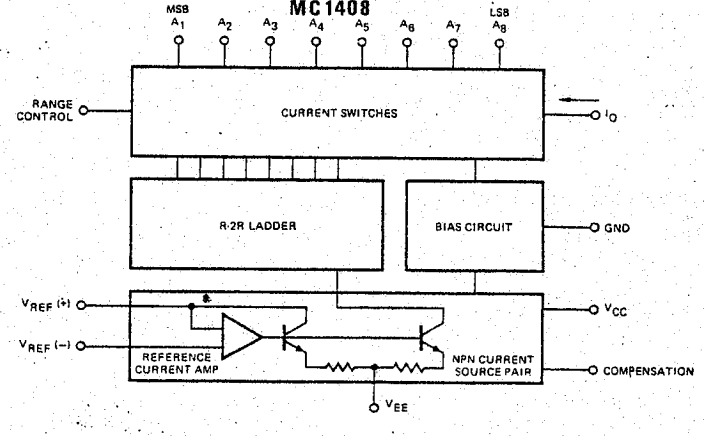
The interface to the microcomputer bus is performed by IC1, an MC6821 Peripherial Interface Adaptor (PIA), which provides two 8-bit bidirectional interface lines and four control lines. The AS-1 uses the "A" I/O section of the PIA (PA0 - PA7) to directly drive the eight data inputs of the MC1408 D/A converter (IC2). The output of the DAC converter is a current which is proportional to the binary inputs. The full scale output current is established by R2 and is nominally 2 ma.

Half of IC3 is used as a current to voltage converter which provides a low impedance output voltage that is proportional to the input current, and therefore the digital input. The scaling of the current to voltage conversion is determined by the sum of the resistances of R4 and R3, which is used to make exact adjustment. C9 provides feedforward compensation for this stage. IC3 is a dual high slew rate op amp parametrically tested and selected for DAC applications.

The "B" I/O port of the PIA (PB0 - PB7) is used to control the A/D circuitry. The input multiplexer, IC5 selects one of 8 analog inputs which are deterimed by the three select inputs. These are connected to the PB0, PB1, and PB2 outputs of the PIA through transistors Q1 to Q3 which act as level translators. Output PB3, using Q4, drives the inhibit input of the analog multiplexer. This disables all inputs from the output, which isolates C3 and provides a sample-and-hold capability.

The second half of IC3 acts as a high input impedance buffer to isolate the output of the multiplexer and minimize the discharge rate of C3 in sample-and-hold mode. IC4 is a precision comparator which has one input connected to the A/D input buffer (unknown voltage) and the other connected to the output of the DAC system. Its output is a digital level that is high when the known voltage is higher than the unknown input voltage to be measured. Its output drives PIA input PB7 which may be sampled by the MPU.

IC7 is a dual tracking current-limited voltage regulator which supplies +9V and -9V to the linear circuitry Its input voltage is derived fron the bus +/- 11-13V. The output current is typically limited to about 30 ma. The value of R19 establishes the nominal output voltage level.



AS-1 PARTS LIST

SEMICONDUCTORS

IC1	MC6821P PIA
IC2	MC1408L8 DAC
IC3	MC1458SP1 DUAL OP AMP
IC4	LM311N COMPARATOR
IC5	CD4051B 8 INPUT ANALOG MULTIPLEXER
IC6	7805 5 VOLT REGULATOR
IC7	MC1468L DUAL REGULATOR
Q1 - Q4	MPS5129 OR 2N5129 NPN TRANSISTOR

CAPACITORS

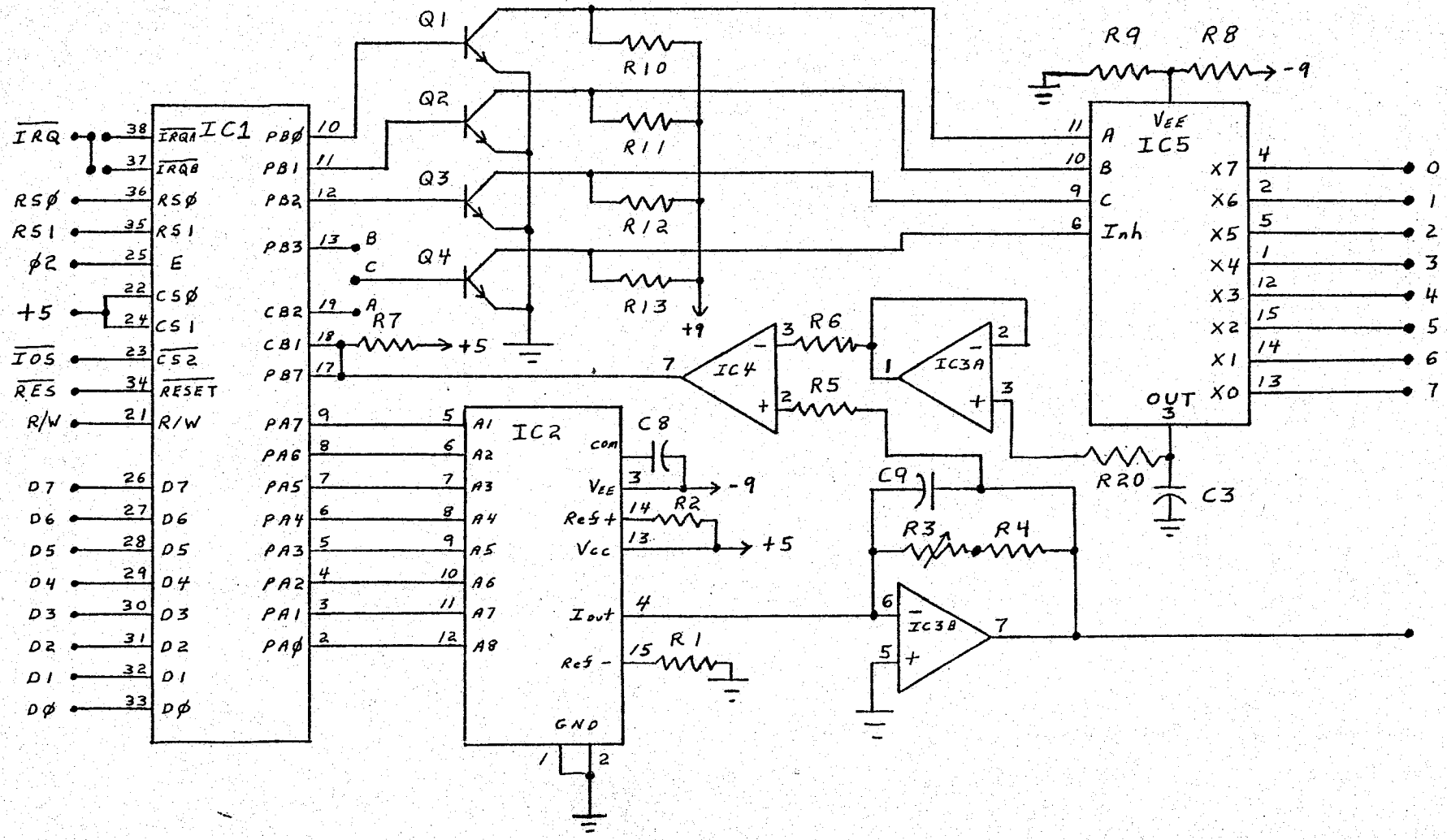
C1,6,7	.1 uF 20V DISC CERAMIC
C2	10 uF 16V ELECTROLYTIC
C3	.01 uF POLYCARBONATE
C4,5	.001 uF DISC CERAMIC
C8	15 pF DISC OR POLYSTYRENE
C9	220 pF DISC CERAMIC

RESISTORS

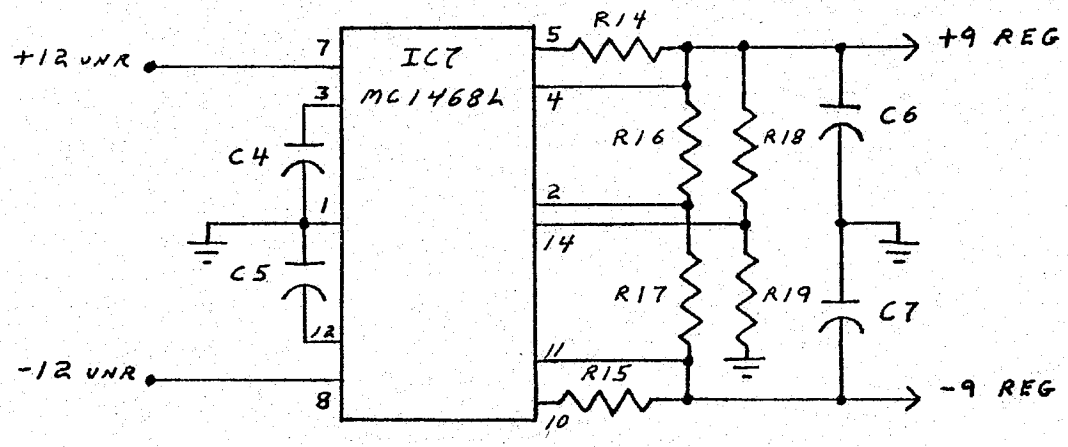
R1,2	2.4K 1/4W
R3	1K TRIMMER
R4	2.7K 1/4W
R5,6	22K 1/4W
R7,10,11,12,13,20	10K 1/4W
R9	470 1/4W
R8	4.7K 1/4W
R14,15	10 1/4W
R16,17	15K 1/4W
R18	1K 1/4W
R19	9.1K 1/4W

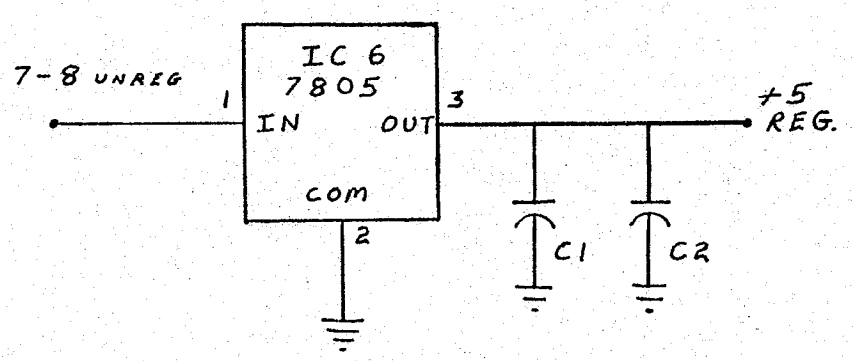
MISCELLANEOUS

QTY	
3	89-01-0010 MOLEX CONNECTOR
1	INDEX PIN
1	4-40 - 1/4 SCREW
1	4-40 NUT
1	#3 LOCKWASHER
1	AS1PC PRINTED CIRCUIT BOARD
1	MANUAL AS1M

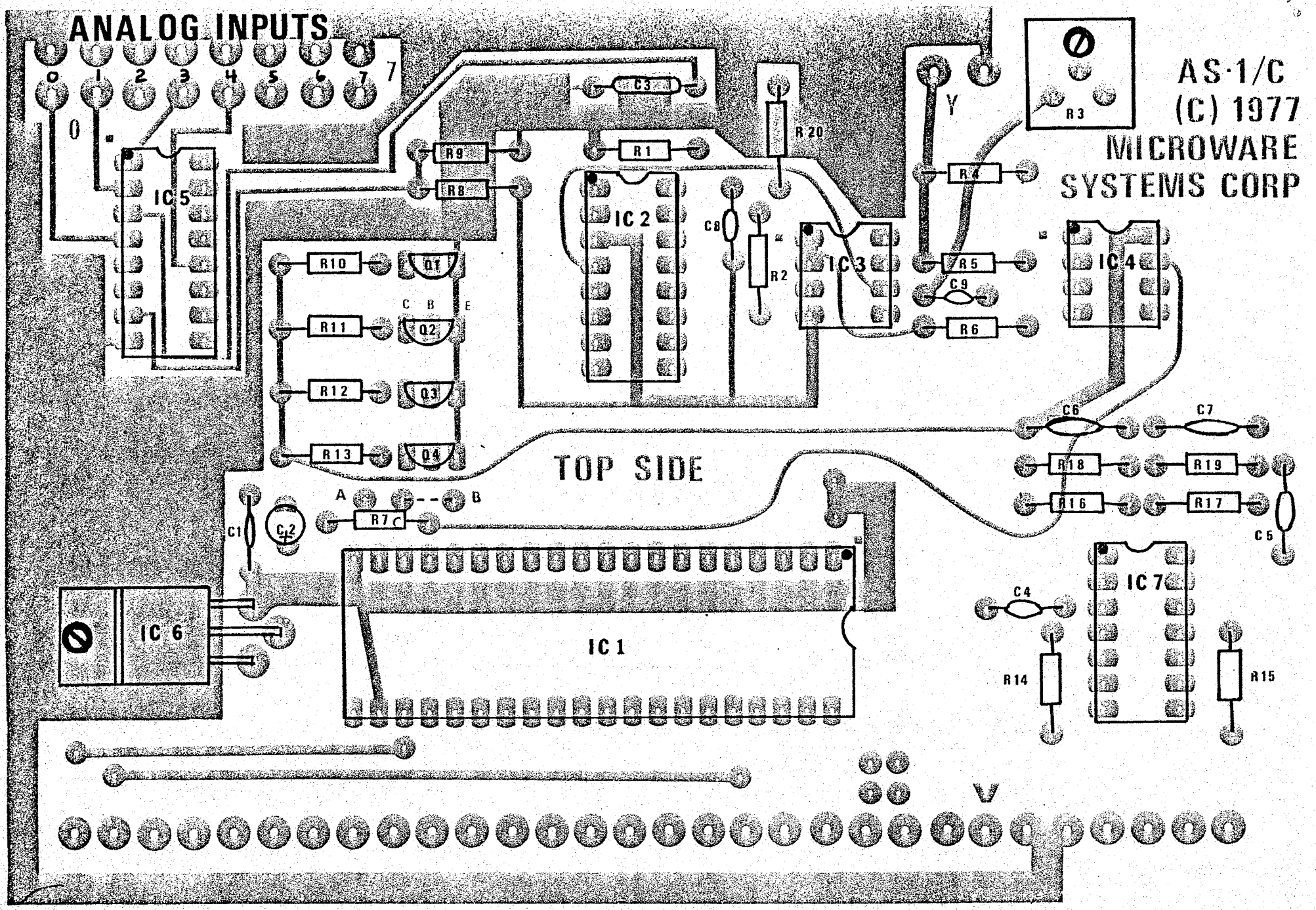


AS1 CIRCUIT DIAGRAM





AS1 POWER SUPPLY


ANALOG INPUTS
0 1 2 3 4 5 6 7
AS-1/C
(C) 1977
MICROWARE
SYSTEMS CORP
IC 1
IC 2
IC 3
IC 4
IC 5
IC 6
IC 7
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13
R14
R15
R16
R17
R18
R19
R20
C1
C2
C3
C4
C5
C6
C7
C8
C9
Q1
Q2
Q3
Q4
TOP SIDE